# Observation scheduling scheme for Subaru telescope

Toshiyuki Sasaki[*a], George Kosugi[a], Jun A. Kawai[b], Toyoaki Kusumoto[b], Norikazu Koura[b], Robert Hawkins[c], Laurence Kramer[c], Anthony P. Krueger[c], and Glenn E. Miller[c]

[a]Subaru Telescope NAOJ, 650 North A'ohoku Place, Hilo, HI 96720
[b]Fujitsu America Inc., 100 Pauahi Street #213, Hilo, HI 96720
[c]Space Telescope Science Institute, 3700 San Martin Drive, Baltimore, MD 21218

## ABSTRACT

Optimization of observation sequences is a function necessary to get high efficiency and reliability of observations. We are now implementing scheduling software into the observation software system for Subaru telescope. The scheduling engine, SPIKE, developed at STScI is used with some modification for the Subaru telescope. An observation target list prepared by observers is converted to SPIKE Lisp codes. SPIKE output is inversely converted to Subaru commands to be executed with the observation software system. Real-time scheduling is planned to re-schedule observations by judging the weather and satisfaction conditions with the help of observation history. The scheduling software can be also used as support tools for observers to indicate an object good for the next observation.

**Keywords:** Observation Scheduling, Real-time Scheduling, Observation Control Software, Subaru telescope

## 1. SUBARU TELESCOPE AND ITS OBSERVATION SOFTWARE SYSTEM

The primary scientific goal of the Subaru telescope is to get astronomical data in the most efficient way as intended with other large telescopes. To achieve this goal and also to avoid diminution of efficiency and safety due to the high altitude at the telescope site atop Mauna Kea, any control operation should be carried out as automatically as possible.

With the Subaru observation software system (SOSS), one can control the telescope and instruments and handle the observation data in coordinated fashion[1,2]. The SOSS is composed of several subsystems for scheduling and logging status, telescope control, observing instrument control, data acquisition, data reduction and analysis software, archival system and database, and remote access control. The SOSS is designed so that the scheduler subsystem controls both the telescope and instruments by actively optimizing the observation procedures. The log subsystem collects status information generated by other subsystems so that the scheduler can recognize every status of the system without access to each subsystem.

A schematic control flow is shown in Figure 1. Basic data input to control systems is *observation procedures* which consist of *observation data sets* (objects and related calibrations) and *observation constraints*[3]. These procedures can be directly executed by a command executor, but in more effective way an *Optimizer* process in the scheduler subsystem arranges observations to get the optimum schedule.

## 2. SCHEDULING SCHEME

Observation Scheduling is considered as key technology to operate the Subaru telescope efficiently and safely[4]. Besides the ultimate goal of automatic scheduling, several benefits are pursued with the scheduler, that is, 1) computer-aided planning of observation proposals and observation procedures, 2) decision support for observers to select the next object by indicating a sequence of observation optimized to some extent, and 3) adequate allocation of observation proposals for the Telescope Allocation Committee.

[*] Correspondence: Email sasaki@subaru.naoj.org; WWW:http://www.naoj.org;

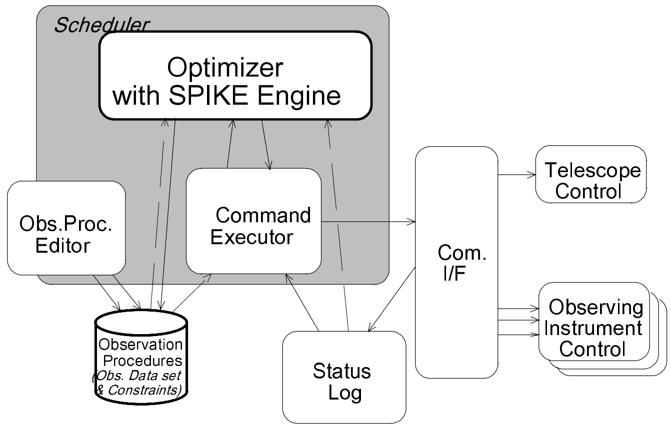


**Figure 1.** A schematic control flow of Subaru Observation Software System.

The scheduling engine, SPIKE, developed at STScI[5] is used with some modification for the Subaru telescope. Based on SPIKE modified for the ESO/VLT[6,7], some extensions were applied to match requests for the Subaru telescope: 1) to minimize redundant calibration (i.e., both external and internal) observations in order to improve efficiency, 2) to add setup times associated with an altitude/elevation change and for a target change for guiding setup time, 3) to distinguish calibrations into external calibration ( standard stars ) and internal calibration ( bias frame, dark frame, and so on ), 4) to

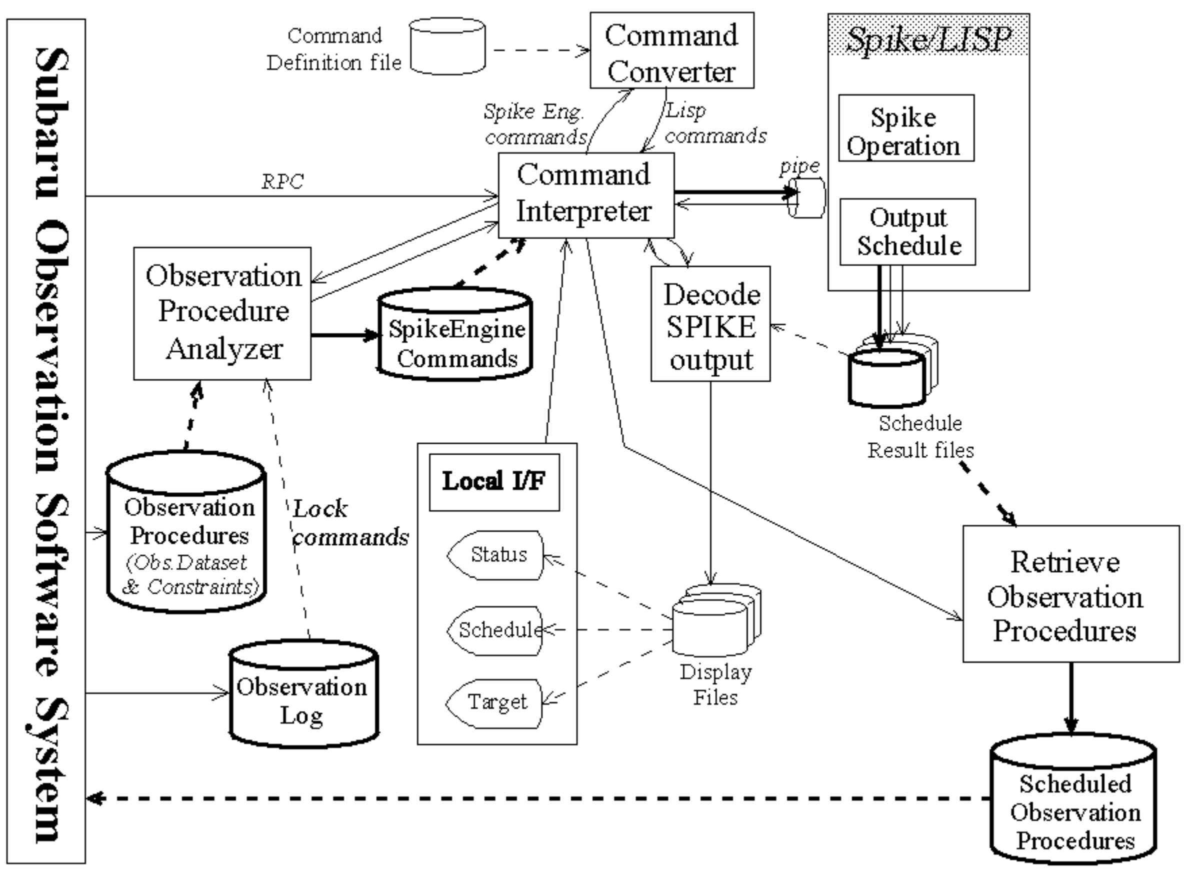


**Figure 2.** A block diagram of the *Optimizer* with SPIKE.

modify format for current-seeing , sky-transparency in suitability mapping, 5) to add individual sun-twilight-angle to override the global default sun-twilight-angle, and 6) to extend a function of no instrument changes for specified time periods other than holidays.

SPIKE is integrated in the *Optimizer* process, which also has an SOSS interface, a SPIKE interface, an observation procedure analyzer, a command converter, a command interpreter for execution, and its local display (Figure 2). An observation target list prepared by observers with some necessary information, such as a instrument name, exposure times, color bands etc., is used to make original observation procedures, which are converted to SPIKE Engine commands by the observation procedure analyzer and then to SPIKE Lisp codes by the command converter. The SPIKE Lisp codes are fed to the SPIKE engine. A SPIKE output is inversely converted to SOSS commands to be executed with the Command Executor process.

In proposed real-time mode during observation, a target list and observation log, from which some targets are identified as observed, are fed to SPIKE to do re-scheduling. As targets observed are locked on its schedule previously determined, we get a new schedule consistent with observations done. With help of frame check methods[8], astronomical frames are examined to have enough calibration frames observed. If not and satisfaction conditions are violated, we need re-scheduling to have reliability of observations. The *Optimizer* should work in the real-time mode by examining the weather and seeing. Weather conditions and seeing size are currently monitored, but a sky monitor is necessary to evaluate the sky transparency.

## 3. CALIBRATION MERGING AND A SCHEDULING EXAMPLE

We added to SPIKE a function to minimize redundant calibration (i.e., both external and internal) frames to achieve high efficiency. We define an observation data set, which consists of a target and related calibration frames. The original observation procedures are composed of a collection of these data sets. Many of the internal calibrations for each target can be merged to a single frame as shown in Figure 3. Some external calibrations, such as standard stars, can also be merged. The current version of SPIKE can merge these redundant calibrations.

The test sample of scheduling with SPIKE implemented with merge function shows that 50 tasks are assigned with 14 object frames and 70 tasks are merged to assigned tasks. As merged tasks are mainly of dark frames of long exposure time and flat-

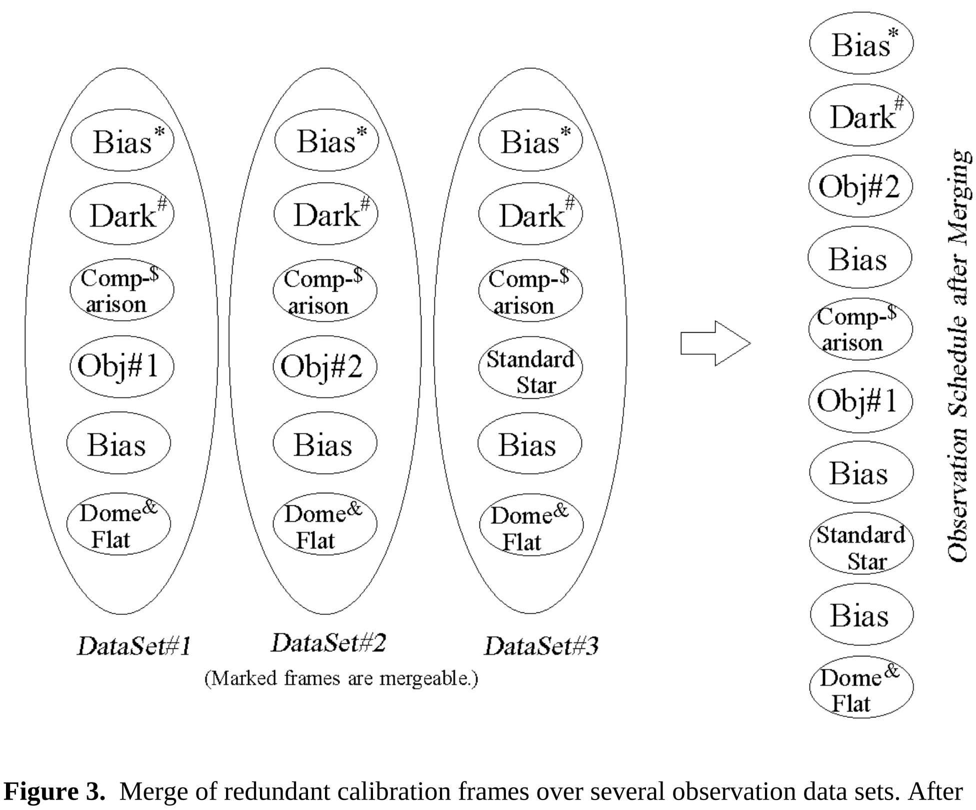


**Figure 3.** Merge of redundant calibration frames over several observation data sets. After merging, three Bias, Dark, Comparison, Dome flat frames are merged to a single one, respectively, in this sample.

fielding frames, observation time are reduced from 20.3 hours to 6.7 hours. Excluding these dark frames and flat-fielding frames, 6 bias frames are merged to reduce 13 min of their read-out time in total since the sample tasks are selected for imaging with no merge condition for standard stars. When we apply the merging function to spectroscopic observation, comparison frames might be merged to reduce much time.

Insertion of Shack-Hartmann (SH) measurement is another function to be implemented in SPIKE. SH is scheduled after a certain time since the last SH done along an observation sequence. It is a good way to adjust the mirror surface and achieve the best focus during the observation. Some time for field setup is also inserted in an observation schedule. When a target is changed, a setup time is necessary for selection of a guide star for an auto-guide.

## 4. GUI FOR SCHEDULING ENGINE

The scheduling engine is integrated in the Subaru Observation Software System. The *Optimizer* has its main GUI window

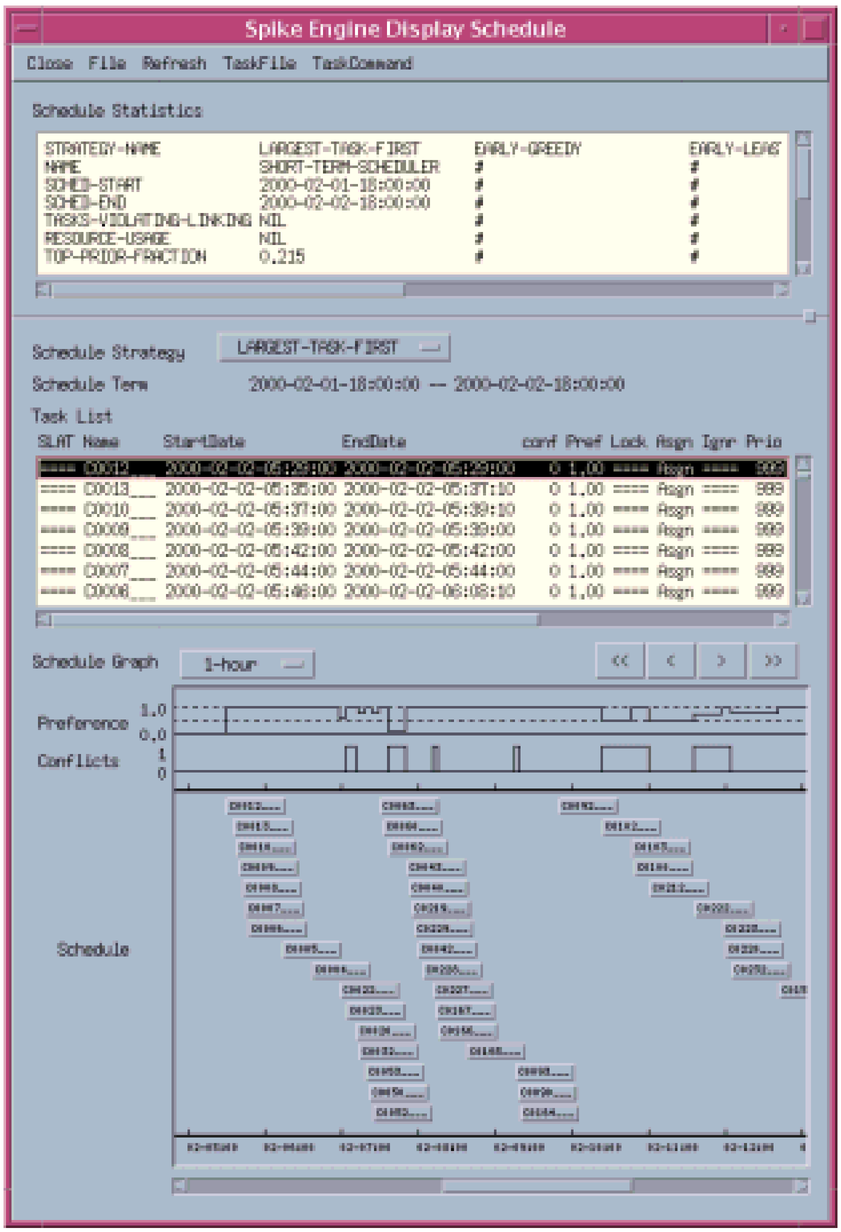


**Figure 4.** Display-schedule window shows a schedule summary, task information and scheduled task display.

which is composed of selection of schedule strategy, a command editor both for SOSS commands and SPIKE commands, and command status. After scheduling is done, a display-schedule window is popped up to display the global schedule results with an individual task information as shown in Figure 4.

## 5. DISCUSSIONS

We are just finished implementing the *Optimizer* with SPIKE. The first trial to get a schedule shows good functionality to merge redundant calibrations. For testing we use the *Optimizer* as support tools for observers to indicate an object good for the next observation.

We are now planning to apply the *Optimizer* in real-time mode during observation. As we don't have the sky monitor yet, continuous monitor of seeing size and completion of command execution may trigger the *Optimizer* to schedule again so that the *Optimizer* will have better functionality for support tools for observers.

## 6. ACKNOWLEDGMENTS

This work is proceeded as a part of Subaru Telescope Software System.

## 7. REFERENCES

1. T. Sasaki, Y. Chikada, R. Ogasawara, S. Ichikawa, W. Tanaka, T. Noguchi, K. Okita, "Concept Design of a Control System for the SUBARU telescope", Nucl. Instr. Methods Phys. Res. A 352, 75, 1994.
2. T. Sasaki, G. Kosugi, J. Noumaru, T. Takata, Y. Mizumoto, R. Ogasawara, Y. Chikada, W. Tanaka, and J. A. Kawai, "Observation control system for SUBARU telescope and its user interface", SPIE 3349, 427, 1998.
3. G. Kosugi, T. Sasaki, Y. Mizumoto, T. Takata, J. A. Kawai, and Y. Ishihara, "Observation Data set of Subaru Observation Software System", SPIE 3349, 421, 1998.
4. T. Sasaki, G. Kosugi, T. Takada, J. Kawai, "Observation Scheduling Scheme of the SUBARU Telescope", Proc. New Observing Modes for the Next Century, ASP Conference Series, Vol.87, 196, 1996.
5. M. D. Johnston and G. E. Miller, "Intelligent Scheduling of Hubble Space Telescope Observations", in Intelligent Scheduling ( eds. M. Zweben and M. S. Fox, Morgan Kaufmann Publ.), Chap.14, p.391,
6. A. M. Chavan, G. Giannone, D. Silva, T. Krueger, and G. E. Miller, "Support tools for the VLT operations: the NTT prototyping experience", SPIE 3349, 97, 1998.
7. A. M. Chavan, G. Giannone, D. Silva, T. Krueger, and G. E. Miller, "Nightly Scheduling of ESO's Very Large Telescope", Astronomical Data Analysis Software and Systems VII, ASP Conference Series, Vol. 145, 255,1998.
8. G. Kosugi, T. Sasaki, Y. Mizumoto, R. Ogasawara, T. Takata, M. Yagi, J. A. Kawai, T. Kusumoto, N. Koura, S. Iwai, "Data quality control of Subaru Telescope", SPIE 4010(this volume), in press, 2000.